

Application No. 2003-332565

[Name of the Document] Request  
[Reference No.] 0390582006  
[Date of Filing] November 24, 2003  
[Address] Commissioner of Patent Office  
5 Mr. Yasuo Imai  
[IPC] H04L 12/00  
[Inventor]  
[Address] c/o Sony Corporation  
7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo  
10 [Name] Takeshi Iwatsu  
[Inventor]  
[Address] c/o Sony Corporation  
7-35, Kitashinagawa 6-chome, Shinagawa-ku, Tokyo  
[Name] Noriyuki Sakoh  
15 [Applicant]  
[ID No.] 000002185  
[Name] Sony Corporation  
[Agent]  
[ID No.] 100082740  
20 [Patent Attorney]  
[Name] Shigemoto Tanabe  
[Indication of Charge]  
[Ledger Account No.] 048253  
[Amount of Payment] 21,000 yen  
25 [List of Document Filed]  
[Document] Scope of Claims 1  
[Document] Specification 1  
[Document] Drawings 1  
[Document] Abstract 1  
30 [General Power of Attorney ID No.] 9709125

[NAME OF DOCUMENT] SCOPE OF CLAIMS

[CLAIM 1]

A method of updating a database schema, characterized by comprising:

5 a requesting step of making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema;

a receiving step of receiving the update information about the program and the update information about the database schema, both transmitted from the update-information providing apparatus in response to the request;

10 a program-updating step of updating the program in accordance with the update information about the program;

a comparing step of comparing, in terms of version, the database schema used by the program with the database schema implemented, based on the update information about the database schema; and

15 a database schema updating step of updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[CLAIM 2]

20 The method of updating a database schema according to claim 1, characterized in that in the database schema updating step, access to the database schema from the program is inhibited while the database schema is being updated.

[CLAIM 3]

25 The method of updating a database schema according to claim 1, characterized in that in the database schema updating step, the database schema is inhibited from being updated while the program is accessing the database schema.

[CLAIM 4]

30 The method of updating a database schema according to claim 1, characterized in that in the database schema updating step, log information is recorded for each command issued to the database schema when the database

schema is updated.

[CLAIM 5]

A communication terminal, characterized by comprising:

5 a requesting means for making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema;

a receiving means of receiving the update information about the program and the update information about the database schema, both transmitted from the update-information providing apparatus in response to the request;

10 a program-updating means for updating the program in accordance with the update information about the program;

a comparing means for comparing, in terms of version, the database schema used by the program with the database schema implemented, based on the update information about the database schema; and

15 a database schema updating means for updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[CLAIM 6]

20 A method of providing update information, comprising:

a storing step of storing update information about a program to be implemented in a communication terminal and update information about a database schema; and

25 a transmitting step of transmitting the update information about the program and the update information about the database schema to the communication terminal when the communication terminal requests the update information about the program and the update information about the database schema,

30 the method being characterized in that the communication terminal is caused to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program

with the database schema implemented, based on the update information about the database schema, and update the database schema implemented based on the update information about the database schema when it is determined from a result of the comparison that the database schema implemented needs to be updated.

5 [CLAIM 7]

An update-information providing apparatus, comprising:

a storing means for storing update information about a program to be implemented in a communication terminal and update information about a database schema; and

10 a transmitting means for transmitting the update information about the program and the update information about the database schema to the communication terminal when the communication terminal requests the update information about the program and the update information about the database schema,

15 the apparatus being characterized in that the communication terminal is caused to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program with the database schema implemented, based on the update information about the database schema, and update the database schema implemented based on the update information about the database schema when it is determined from a result of the comparison that the database schema implemented needs to be updated.

20 [CLAIM 8]

A program for updating a database schema, for causing an information-processing apparatus to execute:

25 a requesting step of making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema;

a receiving step of receiving the update information about the program and the update information about the database schema, both transmitted from the update-information providing apparatus in response to the request;

30 a program-updating step of updating the program in accordance with the

update information about the program;

a comparing step of comparing, in terms of version, the database schema used by the program with the database schema implemented, based on the update information about the database schema; and

5 a database schema updating step of updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[CLAIM 9]

10 A program for providing update information, for causing an information-processing apparatus to execute:

a storing step of storing update information about a program to be implemented in a communication terminal and update information about a database schema; and

15 a transmitting step of transmitting the update information about the program and the update information about the database schema to the communication terminal when the communication terminal requests the update information about the program and the update information about the database schema,

20 the program causing the communication terminal to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program and the database schema implemented based on the update information about the database schema, and update the database schema implemented, based on the update information about  
25 the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[NAME OF DOCUMENT] SPECIFICATION

[TITLE OF THE INVENTION] DATABASE SCHEMA UPDATE METHOD

[FIELD OF THE INVENTION]

[0001]

5           The present invention relates to a method of updating a database schema, and is suitable for use in updating the database schema that is constructed in, for example, a communication terminal.

[BACKGROUND ART]

[0002]

10           Hitherto, update of a database schema performed in order to add tables, columns and the like for storing new data to a database is executed after a manager of the database stops the entire system utilizing the database or switches the system to the backup system.

[0003]

15           Incidentally, in recent years, databases are constructed not only in servers, but also in various communication terminals connected through a network, such as notebook personal computers and PDAs (Personal Digital Assistants). Update of the database schemas constructed in those communication terminals needs to be executed manually by users of the communication terminals.

[0004]

20           In this regard, a method has been proposed, in which a database schema is provided, if necessary, via a network to a communication terminal, and the communication terminal automatically switch the database schema to the new one (see, for example, Patent Document 1).

25                           [Patent Document 1] Japanese Patent Application Laid-open No. 2000-268054

[DISCLOSURE OF THE INVENTION]

[PROBLEMS TO BE SOLVED BY THE INVENTION]

[0005]

30           In this method, however, tables or columns cannot be added while the existing database keeps holding the existing data. After all, a user must

manually update the existing database schema.

[0006]

For that reason, in a case where the user forgets to update the database schema, even if information to be reflected in new tables or new columns has been added to information provided to the communication terminal via the network, for example, the added information cannot be reflected in the tables or columns existing in the database, which is problematic

[0007]

The present invention has been made in view of the foregoing regards, and is intended to propose a method of updating a database schema, which can automatically update the database schema so as to keep a version of the database schema updated.

[MEANS FOR SOLVING THE PROBLEMS]

[0008]

In order to solve the above-mentioned problems, according to the present invention, there is provided a method of updating a database schema, including: a requesting step of making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema; a receiving step of receiving the update information about the program and the update information about the database schema, both transmitted from the update-information providing apparatus in response to the request; a program-updating step of updating the program in accordance with the update information about the program; a comparing step of comparing, in terms of version, the database schema used by the program with the database schema implemented, based on the update information about the database schema; and a database schema updating step of updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[0009]

The version of the database schema implemented is thus recognized from

the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated to be of the latest version all times.

[0010]

5 Further, according to the present invention, there is provided a communication terminal including: a requesting means for making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema; a receiving means of receiving the update information about the program and the update  
10 information about the database schema, both transmitted from the update-information providing apparatus in response to the request; a program-updating means for updating the program in accordance with the update information about the program; a comparing means for comparing, in terms of version, the database schema used by the program with the database schema  
15 implemented, based on the update information about the database schema; and a database schema updating means for updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

20 [0011]

The version of the database schema implemented is thus recognized from the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated to be of the latest version all times.

25 [0012]

Furthermore, according to the present invention, there is provided a method of providing update information, including: a storing step of storing update information about a program to be implemented in a communication terminal and update information about a database schema; and a transmitting step  
30 of transmitting the update information about the program and the update information about the database schema to the communication terminal when the



communication terminal requests the update information about the program and the update information about the database schema, the communication terminal being caused to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program with the database schema implemented, based on the update information about the database schema, and update the database schema implemented based on the update information about the database schema when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[0013]

The version of the database schema implemented is thus recognized from the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated to be of the latest version at all times in compliance with the program.

[0014]

Furthermore, according to the present invention, there is provided an update-information providing apparatus including: a storing means for storing update information about a program to be implemented in a communication terminal and update information about a database schema; and a transmitting means for transmitting the update information about the program and the update information about the database schema to the communication terminal when the communication terminal requests the update information about the program and the update information about the database schema, the communication terminal being caused to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program with the database schema implemented, based on the update information about the database schema, and update the database schema implemented based on the update information about the database schema when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[0015]

The version of the database schema implemented is thus recognized from the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated to be of the latest version at all times.

[0016]

Furthermore, according to the present invention, there is provided a program for updating a database schema, for causing an information-processing apparatus to execute: a requesting step of making a request to an update-information providing apparatus for update information about a program to be implemented and update information about a database schema; a receiving step of receiving the update information about the program and the update information about the database schema, both transmitted from the update-information providing apparatus in response to the request; a program-updating step of updating the program in accordance with the update information about the program; a comparing step of comparing, in terms of version, the database schema used by the program with the database schema implemented, based on the update information about the database schema; and a database schema updating step of updating the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[0017]

The version of the database schema implemented is thus recognized from the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated to be of the latest version at all times.

[0018]

Furthermore, according to the present invention, there is provided a

program for providing update information, for causing an information-processing apparatus to execute: a storing step of storing update information about a program to be implemented in a communication terminal and update information about a database schema; and a transmitting step of transmitting the update information about the program and the update information about the database schema to the communication terminal when the communication terminal requests the update information about the program and the update information about the database schema, the communication terminal being caused to update the program in accordance with the update information about the program, compare in terms of version the database schema used by the program and the database schema implemented based on the update information about the database schema, and update the database schema implemented, based on the update information about the database schema, when it is determined from a result of the comparison that the database schema implemented needs to be updated.

[0019]

The version of the database schema implemented is thus recognized from the update information about the database schema. If the database schema needs to be updated in terms of version, it is automatically updated. Accordingly, the database schema can be automatically updated so as to be of the latest version at all times.

[EFFECT OF THE INVENTION]

[0020]

The present invention can realize a method of updating a database schema, a communication terminal, a method of providing update information, an update-information providing apparatus, a program for updating a database schema, and a program for providing update information, in which the version of the database schema implemented is recognized from the update information about the database schema, and, if the database schema needs to be updated in terms of version, it is automatically updated, whereby the database schema can be automatically updated so as to be of the latest version at all times.

[BEST MODE FOR CARRYING OUT THE INVENTION]

[0021]

Hereinafter, an embodiment of the present invention will be described in detail with reference to the drawings.

[0022]

5 (1) Overall Configuration of Information Providing System

As shown in FIG. 1, reference numeral 1 denotes an information providing system constituting the present invention as a whole, and a client terminal CT receives a broadcast from a radio station RS.

[0023]

10 In the information providing system 1, a related-information providing server KS composed of a computer and connected to the radio station RS by a dedicated line sets up, for the radio station RS, the homepage of the radio station RS. The related-information providing server KS can therefore provide, as related information, music information about the musical pieces broadcast from the radio station RS through a network 2 such as the Internet in response to an acquisition request sent from the client terminal CT.

[0024]

Further, in the information providing system 1, an update-information providing server 3 is connected by way of the network 2. The  
20 update-information providing server 3 provides various update information items such as a program and the like to the client terminal CT via the network 2, in response to a request sent from the client terminal CT.

[0025]

Then, the client terminal CT upgrades its own program on the basis of  
25 the update information provided from the update-information providing server 3.

[0026]

(1-1) Configuration of Radio Station RS

As shown in FIG. 2, in the radio station RS, a control unit 70 formed of a CPU (Central Processing Unit) controls the entirety in accordance with a basic  
30 program such as OS (Operating System) and various application programs, which have been activated via a ROM (Read Only Memory) 71 and a RAM (Random

Access Memory) 72, and a program broadcasting process or the like that broadcasts a program from a program transmitting unit 73 through an antenna 74 to the client terminal CT is performed.

[0027]

5 Further, the radio station RS has a content database 74 that has many music contents to be broadcast in programs and many program contents included in advance. The radio station RS broadcasts the programs as scheduled in a program table and, if necessary, the music contents in the program being broadcast.

10 [0028]

Via a data-communication process unit 76, the radio station RS informs the related-information providing server KS that musical pieces have been broadcast in the program and also informs the related-information providing server KS that which program is being broadcast at present.

15 [0029]

#### (1-2) Configuration of Related-Information Providing Server

As shown in FIG. 3, in the related-information providing server KS, a control unit 90 formed of a CPU controls the entirety and performs specific operations and the like, in accordance with the basic program such as OS and various application programs, which have been activated via a ROM 91 and a RAM 92.

20 [0030]

The related-information providing server KS retrieves, from a music information database 93, the music information related to the musical pieces that the radio station RS has broadcast in the program, and then provides it, as related information, to the client terminal CT via a data-communication process unit 95.

25 [0031]

The related-information providing server KS retrieves, from a program information database 94, the program information related to the program broadcast by the radio station RS, such as the performers' names, the DJ's name, and genres, and provides it to the client terminal CT via the data-communication

30

process unit 95.

[0032]

### (1-3) Configuration of Update-Information Providing Server

As shown in FIG. 4, in the update-information providing server 3, a  
5 control unit 100 formed of a CPU controls the entirety and performs specific  
operations and the like in accordance with the basic program such as OS, an  
update-information providing program, and the like, which have been activated  
via a ROM 101 and a RAM 102.

[0033]

10 The update-information providing server 3 selects an update information  
item from those stored in a hard disk drive 103 in accordance with a request  
provided from the client terminal CT, the selected update information according  
with the request provided from the client terminal CT. The update information  
is provided to the client terminal CT through a data-communication process unit  
15 104.

[0034]

### (1-4) Circuit Configuration of Client Terminal CT

As shown in FIG. 5, a CPU 11 of the client terminal CT controls the  
entirety and performs specific operations and the like in accordance with the basic  
20 program such as OS and various application programs, which have been read  
from a ROM 13 and developed in a RAM 20, the ROM 13 being connected to the  
CPU 11 by a bus 12. The CPU 11 performs communication behavior via, for  
example, the network 2, input and output operations from and to the user,  
playback of contents from media, write of the contents downloaded from the  
25 radio station RS into a hard disk drive (HDD) 21, management of these contents,  
and the like.

[0035]

An operation input unit 15 sends, to an input process unit 14, information  
that the user has input by operating various operators provided on the surface of a  
30 main-unit housing or on a remote controller (not shown). The input process unit  
14 performs a prescribed process on this information and then sends it to the CPU

11 as an operation command, and the CPU 11 performs a process that accords with the operation command.

[0036]

5 A display 17 may be a display device, such as a liquid crystal display, provided directly on the surface of the main-unit housing, or may be an external display device connected to the main-unit housing. The display 17 displays the results of processes performed by the CPU 11 and various information items.

[0037]

10 A media drive 19 is a drive that plays back, for example, CDs (Compact Discs) or memory sticks (trademark) formed of a flash memory or the like. The played-back result is subjected to digital-to-analog conversion via an audio data process unit 24, and then is output from a 2-channel speaker 25.

[0038]

15 It should be noted that the CPU 11 can store the data played back via the media drive 19, as an audio file, into the hard disk drive 21 if the data is the audio content of a musical piece.

[0039]

20 Further, the CPU 11 can cause the media drive 19 to read a plurality of still pictures stored in the memory stick, and can perform a slide show, displaying them via a display process unit 16 on the display 17.

[0040]

25 Moreover, the CPU 11 can make random access to the hard disk drive 21, thereby to read a plurality of music information items stored in the drive 21. Therefore, the music information items can be played back and output in an order requested by the user as if they were played by a jukebox.

[0041]

30 A tuner unit 27 is, for example, an AM or FM radio tuner. The tuner unit 27 demodulates the broadcast signal received by the antenna 26 in accordance with the control of the CPU 11, and outputs the resultant, as broadcast audio signal, via the audio data process unit 24 from a speaker 25.

[0042]

A communication process unit 22 encodes the data to be transmitted, under the control of the CPU 11 and transmits it through a network interface 23 via the network 2 to an external apparatus that is compatible with the network. The communication process unit 22 also decodes data received from the external apparatus that is compatible with the network via the network interface 23 and transfers it to the CPU 11.

[0043]

#### (1-5) Directory Management of Contents

The CPU 11 of the client terminal CT manages the contents by the directory configuration illustrated in FIG. 6 when storing the contents in the hard disk drive 21. First, "folder" directories, as lower layers of a "root" directory, are formed in a given number falling within a prescribed range. The "folder" directories are formed to correspond to genres of contents or users owning the contents, respectively.

[0044]

Then, "album" directories, as lower layers of the "folder" directories, are formed in a given number within a prescribed range. Each of the "album" directories corresponds to, for example, one album title. One or more "track" files belonging to an "album" directory are stored in lower layers of the "album" directory. Each "track" file is one musical piece, i.e., content.

[0045]

The directory management of such contents is carried out with database files that are stored in the hard disk drive 21.

[0046]

#### (1-6) Program Module Configuration of Client Terminal CT

As shown in FIG. 7, a program module of the client terminal CT operates in accordance with the OS. More specifically, the module performs data exchange between a CD seller server 31 in which CDs are sold, an Internet radio server 32, a music provider server 33, an integrated service server 34, the related-information providing server KS, and other various servers. It should be noted that the integrated service server 34 includes the update-information



providing server 3 described above.

[0047]

An HTTP (Hyper Text Transfer Protocol) message program 36 achieves HTTP communication between the CD seller server 31, the Internet radio server 32, the music provider server 33, the integrated service server 34, the related-information providing server KS and other various servers. A communicator program 37 is a program module that accomplishes data exchange with the HTTP message program 36.

[0048]

In a layer above the communicator program 37, a content playback module 38 and a copyright protection information management module 39 are located, the content playback module 38 interpreting the codec of content and playing back the content, the copyright protection information management module 39 handling information concerning copyright. An Internet radio-station selection/playback module 43 is provided to select an Internet radio station and play back the Internet radio, for the content playback module 38. A musical-piece purchase/playback module 44 is provided to purchase any musical piece and play back the musical piece of sample, for the copyright protection information management module 39.

[0049]

In a layer that lies above the Internet radio-station selection/playback module 43 and the musical-piece purchase/playback module 44, an XML (eXtensible Markup Language) browser 50 is located. The XML browser 50 interprets XML files supplied from various servers and performs display on the display 17.

[0050]

For example, a musical piece selected by the user via the XML browser 50 is purchased in the musical-piece purchase/playback module 44, and the musical piece thus purchased is written into the hard disk drive 21 through a hard-disk content controller 42.

[0051]

It should be noted that an authentication library 47A of a library 47 is connected to the communicator program 37. The authentication library 47A performs an authentication process on the integrated service server 34 and the other various servers.

5 [0052]

Further, in a layer above the communicator program 37, there are located a database access module 40, a content-data access module 41, and the hard-disk content controller 42.

10 [0053]

The database access module 40 can access the various databases constructed in the hard disk drive 21. The content-data access module 41 can access the contents stored in the hard disk drive 21. The hard-disk content controller 42 manages the contents stored in the hard disk drive 21.

15 [0054]

In a layer above the hard-disk content controller 42, there are located a related-information display module 45 and a tuner-selection playback/recording module 46, the related-information display module 45 being designed to display the titles of the musical pieces broadcast by the radio station RS and the names of the artists, the tuner-selection playback/recording module 46 selecting the radio station RS and recording in the hard disk drive 21 the musical contents received from the radio station RS.

20 [0055]

For example, a musical piece received from the radio station RS that has been selected via an audio user interface 51 is written into the hard disk drive 21 through the content-data access module 41.

25 [0056]

The related-information display module 45 receives the titles and artists' names of the musical pieces being broadcast by the radio station RS by the tuner-selection playback/recording module 46, as related information, from the related-information providing server KS via the HTTP message 36, and displays them on the display 17 via the audio user interface (UI) 51.

30

[0057]

It should be noted that the related information displayed on the display 17 via the audio user interface 51 can be temporarily stored in a clip library 47B of the library 47, and it can be finally stored into the hard disk drive 21 via the database access module 40 in accordance with an instruction of the user.

[0058]

The client terminal CT further includes program modules, i.e., a CD playback module 48 for playing back CDs and an HDD playback module 49 for playing back the hard disk drive 21. The data played back is output via the audio data process unit 24 and the speaker 25.

[0059]

(2) Acquisition Sequence for Related Information

Next, an acquisition sequence in which the client terminal CT acquires, from the related-information providing server KS, as related information, information about musical pieces broadcast in a program of the radio station RS will be explained with reference to FIG. 8.

[0060]

In Step SP1, the radio station RS starts broadcasting a current program. In Step SP2, when the station RS starts broadcasting a musical piece A in the current program, the related-information providing server KS is informed of that the broadcast of the musical piece A is started.

[0061]

At this time, in Step SP11, the related-information providing server KS holds, as a database, in the music information database 93 (FIG. 3), the music information about musical pieces, such as the titles and the names of artists related to various types of musical pieces, and images printed on CD jackets containing the musical pieces. Because the music information related to the musical piece A informed from the radio station RS is provided to the client terminal CT, the related-information providing server KS updates the previously-held music information to the music information related to the musical piece A. Then, the process goes to Step SP12.

[0062]

Incidentally, in Step SP21, the client terminal CT receives the program that the radio station RS broadcasts in Step SP1 and the musical piece A broadcast in the program, too. Then, the process goes to Step SP22.

5 [0063]

In Step SP22, the client terminal CT transmits an acquisition request for the music information about the musical piece A, such as the title and the artist's name, to the related-information providing server KS. The process then goes to Step SP23.

10 [0064]

At this time, in response to the acquisition request from the client terminal CT, the related-information providing server KS provides, in Step SP12, the music information about the musical piece A, which is currently held to be provided, to the client terminal CT via the network 2.

15 [0065]

Accordingly, the client terminal CT causes the display 17 to display, as related information, the music information about the musical piece A, which has been provided from the related-information providing server KS, with the result that the user can recognize as related information the music information about the musical piece A broadcast in the program, such as the title and the artist's name.

20 [0066]

In Step SP23, the client terminal CT stores the music information about the musical piece A, provided from the related-information providing server KS, in the database constructed in the hard disk drive 21 as related information, with the result that the client terminal CT can thereafter read any music information as demanded by the user from the database and display the read information on the display 17.

[0067]

### (3) Schema Changes in Database

30 Incidentally, in the information providing system 1, when the entire information providing system 1 or part thereof is upgraded, the

related-information providing server KS may provide related information containing additional items that are different from before, along with the upgrade.

[0068]

5 For example, before the upgrade, the music information provided from the related-information providing server KS to the client terminal CT contains only a title and an artist's name of a musical piece and a jacket image of a CD containing the musical piece. On the other hand, after the upgrade, words and a CD number of the musical piece are newly added to the music information.

[0069]

10 In this case, the programs and schema of the database need to be upgraded in the client terminal CT so that the newly added items may also be displayed on the display 17 and recorded in the database, for example.

[0070]

15 Here, as shown in FIG. 9, the word "programs" means the various applications that work on the OS of the client terminal CT and the program module illustrated in FIG. 6. The various applications, which become the uppermost layer of the programs, perform communication with the OS via the program module, which is the lowermost layer thereof. Thus, the display 17 can display desired information, the speaker 25 can output desired sound, and an access can be made to the database constructed in the hard disk drive 21.

[0071]

20 Actually, the CPU 11 of the client terminal CT accesses the update-information providing server 3 and checks whether its own programs are of the latest version. If its own programs are not of the latest version, the CPU 11 makes a request to the update-information providing server 3 for transmission of the latest programs as update information.

[0072]

30 When the update-information providing server 3 receives the request for the transmission of the latest programs from the client terminal CT, it transmits the aforementioned latest programs to the client terminal CT in response to the request. At this time, together with the latest programs, the update-information

providing server 3 transmits, to the client terminal CT, a schema-change instruction file SF, which will be described later, for making the database schema of the client terminal CT comply with the aforementioned latest programs.

[0073]

5           When the CPU 11 of the client terminal CT finishes receiving the latest programs and the schema-change instruction file SF, it records the schema-change instruction file SF in the hard disk drive 21, and upgrades its own programs to the latest programs.

[0074]

10           After upgrading the programs, the CPU 11 starts upgrading the schema of the database. Here, the client terminal CT has the function of automatically change the schema of the database in accordance with the schema-change instruction file SF provided from the update-information providing server 3. The schema-changing function will be explained below.

15           [0075]

FIG. 10 shows the outline of a schema-changing process in the schema-changing function. After upgrading the programs, the CPU 11 of the client terminal CT carries out a system-booting process SB via the OS, the system-booting process SB booting the entire system of the client terminal CT (hereinafter called terminal system). After carrying out the system-booting process SB, the CPU 11 of the client terminal CT starts a schema-changing process SC via the database access module 40 in Step Ar1. It should be noted that, in the following description, the schema-changing process SC is carried out by the CPU 11 via the database access module 40 that is one of the program modules.

25           [0076]

When the schema-changing process SC is started, the database DB constructed in the hard disk drive 21 is activated in Step Ar2. The process then goes to Step Ar3.

30           [0077]

In Step Ar3, the schema-changing process SC reads the schema-change

instruction file SF provided from the update-information providing server 3 from the hard disk drive 21, and then goes to Step Ar4.

[0078]

5 In Step Ar4, the schema-changing process SC issues an SQL command described in the schema-change instruction file SF, which will be described later, to the database DB. The schema of the database DB is therefore changed. Then, the schema-changing process SC goes to Step Ar5.

[0079]

10 In Step Ar5, the schema-changing process SC writes log information about the SQL command issued in Step Ar4 into a schema-change log file SL, which will be described later. The schema-changing process SC is then terminated.

[0080]

15 The CPU 11 is designed to execute the schema-changing process SC as described above via the database access module 40, after upgrading the programs.

[0081]

20 Here, the schema-changing process SC will be described in more detail, with reference to the flowchart shown in FIG. 11. It should be noted that Steps SP30 to SP37 shown in the flowchart of FIG. 11 correspond to Steps Ar2 to Ar5 shown in FIG. 10.

[0082]

After carrying out the system-booting process SB via the OS, the CPU 11 carries out a schema-change procedure RT10 in accordance with a schema-changing program that has been read from the ROM 13.

25 [0083]

Upon carrying out the schema-change procedure RT10, the CPU 11 activates the database DB in Step SP30. The procedure then goes to Step SP31.

[0084]

30 In Step SP31, the CPU 11 reads the schema-change instruction file SF from the hard disk drive 21. The procedure then advances to Step SP32.

[0085]

As shown in FIG. 12, the schema-change instruction file SF is a file of text format, including a plurality of lines, and is constituted of a comment part cmt in which each line starts with "#", and a command parts cmd in which every line starts with a 3-digit number and a 2-digit number. It should be noted that in the last line of the command parts cmd, only EOF (End Of File) indicating the end position of the schema-change instruction file SF is described.

[0086]

In each line of the comment part cmt, a comment sentence with respect to the schema-change instruction file SF is written, following the mark "#".

[0087]

In every line of the command part cmd (except the last line), mark ":" is added after the 3-digit and 2-digit numbers, and an SQL command for changing the schema follows the mark ":". Here, the SQL command is adapted so that only a command is described for one line and a command is not described over a plurality of lines.

[0088]

Further, as to the numbers SD (hereinafter called indicator information), i.e., 3-digit number and 2-digit number, which are described at the head of each line of the command part cmd, the preceding 3-digit number represents the version of the schema, and the following 2-digit number represents the process number in the schema-change instruction file SF for the version.

[0089]

That is, if the indicator information SD is assumed to be "002-02", for example, the schema is of version "002", and the process number in the schema-change instruction file SF for version "002" is "02".

[0090]

In practice, for example in the schema-change instruction file SF shown in FIG. 12, there are described "002-01: alter table MusicTable add (rylics varchar (256));" in the first line of the command part cmd, "002-02: create table TOCTable (tocid int, tocinfo byte (804));" in the second line thereof, "002-03: alter table MusicTable add foreign key (tocid) references TOCTable;" in the third



line thereof, and EOF in the fourth line thereof. In this case, the CPU 11 issues three SQL commands described in each line, one by one in the order of the process numbers.

[0091]

5 In Step SP32 (FIG. 11), the CPU 11 reads the schema-change log file SL that has been recorded in the hard disk drive 21 when the upgrade was carried out before. Here, the schema-change log file SL is a file of text format, consisting of one line only, as shown in FIG. 13. In this line, a 3-digit number and 2-digit number (hereinafter, called indicator information log) SDL is described as log  
10 information.

[0092]

The indicator information log SDL corresponds to the indicator information SD of the schema-change instruction file SF. In practice, every time the SQL command of each line described in the command part cmd of the  
15 schema-change instruction file SF is issued to the database DB, the schema-change log file SL is updated with the indicator information SD of the line. The CPU 11 can therefore recognize to which process number of which version the schema-changing process has been finished by acquiring the indicator information log SDL of the schema-change log file SL.

20 [0093]

After reading the schema-change log file SL in Step SP32, thus acquiring the indicator information log SDL thereof, the CPU 11 goes to the next step, i.e., Step SP33.

[0094]

25 In Step SP33, the CPU 11 reads one line of the command part cmd of the schema-change instruction file SF, and then goes to the next step, i.e., Step SP34.

[0095]

In Step SP34, the CPU 11 judges whether EOF is described in the line that has been read. If the judgment made here is negative, this means that the  
30 line read is not the last line of the schema-change instruction file SF, but is a line that describes an SQL command. In this case, the CPU 11 goes to the next step,

i.e., Step SP35.

[0096]

In Step SP35, the CPU 11 compares the indicator information SD of the read line with the indicator information log SDL described in the schema-change log file SL. Thus, the CPU 11 judges whether the indicator information SD is  
5 smaller than the indicator information log SDL.

[0097]

When the judgment made here is affirmative, this means a case where the indicator information of the schema-change log file SL is, for example,  
10 "002-02" and the indicator information SD of the line read from the schema-change instruction file SF is, for example, "002-01", and means that the SQL command described in the line has been issued to the database DB. Therefore, the CPU 11 returns to Step SP33 and reads the next line.

[0098]

15 Meanwhile, when the judgment made in Step SP35 may be negative, this means that the SQL command described in the read line has not been issued. Then, the CPU 11 goes to the next step, i.e., Step SP 36.

[0099]

In Step SP36, the CPU 11 issues the SQL command described in the read  
20 line to the database DB. The CPU 11 thus changes the schema of the database DB and goes to the next step, i.e., SP37.

[0100]

In Step SP37, the CPU 11 uses the indicator information SD of the read line to update the indicator information log SDL of the schema-change log file SL.  
25 Then, the CPU 11 returns to Step SP33 and reads the next line.

[0101]

Thus, by repeating the processes of Steps SP33 to SP37 until EOF is acquired in Step SP34, the CPU 11 issues the SQL commands described in the command part cmd of the schema-change instruction file SF, line by line, and  
30 changes the schema of the database DB. Then, the CPU 11 goes to the next step, i.e., SP 38, terminating the schema-change procedure RT10.

[0102]

As a result, as shown in FIG. 14 for example, the CPU 11 issues the SQL commands described in the schema-change instruction file SF to the database DB in which the schema is formed of only the table "MusicTable" composed of three columns "title", "artist" and "coverart" before the upgrade, and thus adds columns "rylics" and "tocid" to the table "MusicTable" and generates a new table "TOCTable" composed of two columns "tocid" and "tocinfo". Thus, the database DB can be automatically upgraded to a database having a schema of a structure in which the column "tocid" of the table "MusicTable" is linked to the column "tocid" of the table "TOCTable".

[0103]

The schema of the database DB is thus automatically upgrade in conformity with the programs on the basis of the schema-change instruction file SF. Hence, when related information that has additional items different from before due to the version-up of the information providing system 1 is provided from the related-information providing server KS, the related information can be reflected in the database DB.

[0104]

Further, every time the CPU 11 issues the SQL commands described in each line of the schema-change instruction file SF to the database, it updates the schema-change log file SL by using the indicator information SD of that line as indicator information log SDL. Accordingly, even if the user forcefully terminates the terminal system while the SQL command in a certain line is being issued or the terminal system is shut down due to power failure, for example, the issuance of the SQL command in the interrupted line can be resumed on the basis of the indicator information log SDL of the schema-change log file SL when the schema-changing process SC is carried out again. This can prevent double issuance of the SQL command of the same version.

[0105]

(4) Exclusive Control Process

Moreover, the client terminal CT has an exclusive control function of

exclusively controlling the access to the database DB during the execution of the schema-changing process SC or during the execution of a specific application. An exclusive control process performed during the schema-changing process and an exclusive control process performed during the execution of the application will be separately explained below.

[0106]

#### (4-1) Exclusive Control Process During Schema-Changing Process

FIG. 15 shows the outline of the exclusive control process performed during the schema-changing process. After the schema-changing process SC is started as described above, in practice, the CPU 11 of the client terminal CT starts an exclusive control process HS via the OS in Step Ar10 to lock the database DB. With this, a predetermined application AP that executes processes other than the schema-changing process SC cannot access the database DB until the database DB is unlocked. It should be noted that the application AP is executed by the CPU 11 via a program module and an exclusive control process HS is performed by the CPU 11 via the OS.

[0107]

After the database DB is locked, the schema-changing process SC proceeds to Step Ar11. In Step Ar11, the schema of the database DB is changed as described above.

[0108]

At this time, the exclusive control process HS prohibits the application AP from accessing the database DB. Therefore, in a case where the application AP tries to make access to the database DB in Step Ar12, for example, it remains in a waiting state for start of the process until the database DB is unlocked.

[0109]

When the schema of the database DB is completely changed, the schema-changing process SC recognizes it in Step Ar13 and goes to the next Step Ar14 to cause the exclusive control process HS to unlock the database DB.

[0110]

On unlocking the database DB, the exclusive control process HS makes

the process of the application AP start, the application AP being in the waiting state in Step Ar15. The application AP then goes to Step Ar16 and makes access to the database DB.

[0111]

5 As described above, in the client terminal CT, the exclusive control process HS is executed so that the application AP cannot access the database DB during the execution of the schema-changing process SC.

[0112]

10 Here, the exclusive control process HS will be described in more detail with reference to the arrow chart of FIG. 16. It should be noted that Steps S100 to S110 in the arrow chart shown in FIG. 16 correspond to Steps Ar10 to Ar14 shown in FIG. 15.

[0113]

15 The CPU 11 boots up the system via the OS after it has updated the programs in terms of version. In Step S100, the schema-changing process SC is started via the database access module 40.

[0114]

20 The schema-changing process SC thus started starts the exclusive control process HS in Step S101, and at the same time, it requests the exclusive control process HS to lock the database DB.

[0115]

25 When the exclusive control process HS is requested to lock the database DB from the schema-changing process SC, it locks the database DB in Step S102 in response to the request and sends an acceptance of the lock back to the schema-changing process SC.

[0116]

30 On recognizing the acceptance of the lock from the exclusive control process HS, the schema-changing process SC activates the database DB in Step S103. The schema-changing process SC then goes to the next step, i.e., Step S104.

[0117]

In Step S104, the schema-changing process SC makes a request to the database DB for connection.

[0118]

On recognizing the connection request coming from the schema-changing process HS, the database DB starts the connection to the schema-changing process SC. In Step S105, the database DB then informs the schema-changing process SC that the connection has been initiated.

[0119]

On recognizing the connection informed by the database DB, the schema-changing process SC goes to Step S106. In Step S106, the schema-changing process SC reads the schema-change instruction file SF and changes the schema of the database DB in accordance with the instruction of the schema-change instruction file SF. Further, the schema-changing process SC performs a series of the schema-changing processes (FIG. 10 and FIG. 11) for updating the schema-change log file SL, and then goes to the next step, i.e., Step S107.

[0120]

In Step S107, the schema-changing process SC requests disconnection from the database DB.

[0121]

On recognizing the request for disconnection coming from the schema-changing process SC, the database DB disconnects the schema-changing process SC therefrom, and in Step S108, informs the schema-changing process SC that it has been disconnected from the schema-changing process SC.

[0122]

On recognizing the information of the disconnection from the database DB, the schema-changing process SC goes to Step S109. In Step S109, the schema-changing process SC requests that the exclusive control process HS unlock the database DB.

[0123]

On recognizing the unlock request coming from the schema-changing

process SC, the exclusive control process HS unlocks the database DB in response to the request in Step S110, and sends the acceptance of the unlock back to the schema-changing process SC.

[0124]

5           On recognizing the acceptance of the unlock coming from the exclusive control process HS, the schema-changing process SC informs the system of the termination of the schema-changing process SC in Step S111.

[0125]

10           As described above, in the client terminal CT, the exclusive control process HS is executed so that any other application AP is inhibited from accessing the database DB while the schema-changing process SC is being executed. The data stored in the database DB can therefore remain in consistency.

[0126]

15           (4-2) Exclusive Control Process in Application Execution

Next, the exclusive control process performed while the application is being executed will be described with reference to the arrow chart of FIG. 17.

[0127]

20           Specifically, if the application AP is executed after the programs and the database DB have been updated in terms of version, it requests in Step S120 that the exclusive control process HS lock the database DB.

[0128]

25           When requested to lock the database DB by the application AP, the exclusive control process HS locks the database DB in Step S121, and then sends the lock acceptance back to the application AP.

[0129]

On recognizing the lock acceptance sent from the exclusive control process HS, the application AP requests connection from the database DB.

[0130]

30           On recognizing the connection request coming from the application AP, the database DB starts the connection to the application AP and then informs the

application AP that the connection has been initiated in Step S123.

[0131]

On recognizing the connection informed by the database DB, the application AP goes to Step S124. In Step S124, the application AP issues a desired SQL command to the database DB, with the result that desired related information can be stored in or read from the database DB.

[0132]

After issuing the desired SQL command, the application AP goes to Step S125 and in Step S125, requests that the database DB be disconnected therefrom.

[0133]

On recognizing the request for disconnection coming from the application AP, the database DB disconnects the application AP therefrom and in Step S126, informs the disconnection to the application AP.

[0134]

On recognizing the disconnection from the database DB, the application AP goes to Step S127 and in Step S127, requests that the exclusive control process HS unlock the database DB.

[0135]

On recognizing the unlock request coming from the application AP, the exclusive control process HS unlocks the database DB in response to the request in Step S128, and sends the acceptance of the unlock back to the application AP.

[0136]

As described above, in the client terminal CT, the exclusive control process HS is executed so that any other application AP and the schema-changing process SC are inhibited from simultaneously accessing the database DB while the application AP is accessing the database DB. Accordingly, the data stored in the database DB can therefore remain in consistency, which prevents unauthenticated data from being stored or the data from being erased by error.

[0137]

#### (5) Operation and Effects

In the configuration described above, the client terminal CT requests, if



its own programs are not of the latest version, that the update-information providing server 3 transmit the programs of the latest version and the schema-change instruction file SF.

[0138]

5 In response to the request coming from the client terminal CT, the update-information providing server 3 transmits the programs of the latest version and the schema-change instruction file SF to the client terminal CT. After receiving the programs of the latest version and the schema-change instruction file SF, the client terminal CT records the schema-change instruction file SF in  
10 the hard disk drive 21 and updates its own programs in terms of version.

[0139]

After upgrading the programs, the client terminal CT proceeds to an upgrade process for the database schema. The client terminal CT reads the schema-change instruction file SF, and at the same time, reads the schema-change  
15 log file SL that was recorded in the hard disk drive 21 when the database schema was updated in terms of version before.

[0140]

The client terminal CT compares the indicator information SD described in each line of the schema-change instruction file SF with the indicator  
20 information log SDL described in the schema-change log file SL updated at the previous version-up. Then, the client terminal CT issues, to the database DB, only the SQL command described in the line that is an indicator information SD greater than the indicator information log SDL command by command in the order of the indicator information DS, and updates the schema-change log file SL  
25 with the indicator information SD.

[0141]

As described above, in the client terminal CT, the schema-change instruction file SF and the schema-change log file SL are compared, whereby the version of the current schema is recognized and is automatically updated if the  
30 current schema is not of the latest version.

[0142]

Thus, the client terminal CT can automatically update the schema of the database DB so that the schema thereof is always of the latest version.

[0143]

Accordingly, all related information can reliably be reflected in the database DB when the related-information providing server KS provides the related information to which more items are added than before due to the version-up of the information providing system 1. Hence, the related information can be reliably prevented from leaking.

[0144]

Further, in the client terminal CT, the exclusive control process HS is executed so that any other application AP is inhibited from accessing the database DB while the schema-changing process SC is being carried out. The data stored in the database DB can therefore remain in consistency, which prevents unauthenticated data from being stored or the data from being erased by error.

[0145]

Moreover, in the client terminal CT, the exclusive control process HS is executed so that the schema-changing process SC for the database DB or any other application AP is inhibited from accessing the database DB while the application AP is being executed. The data stored in the database DB can therefore remain in consistency, which prevents unauthenticated data from being stored or the data from being erased by error.

[0146]

In the configuration described above, the client terminal CT receives the programs of the latest version and the schema-change instruction file SF from the update-information providing server 3 to upgrade the programs, and then compares the indicator information SD of the schema-change instruction file SF with the indicator information SDL of the schema-change log file SL to recognize the version of the present schema. If it is judged that the present schema is not of the latest version, the schema is automatically changed based on the schema-change instruction file SF. Thus, the database schema can be automatically updated to be of the latest version all times.

[0147]

(6) Other Embodiments

It should be noted that in the embodiment described above, there has been described the case where the database DB constructed in the client terminal CT is automatically updated in terms of version, but the present invention is not limited to this. Various communication terminals connected to the update-information providing server 3 via the network 2 and the database DB constructed in the server may be automatically updated in terms of version.

[0148]

Further, in the embodiment described above, there has been described the case where the programs of the latest version are provided as program update information to the client terminal CT, but the present invention is not limited to this. Patches or the like for upgrading the programs may be provided as program update information to the client terminal CT.

[0149]

Moreover, in the embodiment described above, there has been described the case where the schema-change instruction file SF as update information is a file of text format, but the present invention is not limited to this. Any other file of various formats may be used as long as the client terminal CT can recognize it.

[0150]

Further, in the embodiment described above, there has been described the case where the schema-change log file SL as log information of the schema-changing process SC is a file of text format, but the present invention is not limited to this. Any other file of various formats may be used as long as the client terminal CT can recognize it.

[0151]

In this case, a schema-change log table may be generated in place of, for example, the schema-change log file SL, and the indicator information log SDL may then be written in the schema-change log table.

[0152]

Further, in the embodiment described above, there has been described

the case where the CPU 11 of the client terminal CT develops in the RAM 20 the schema-changing program stored in ROM 13 in advance, and the above-mentioned schema-changing process SC is executed in accordance with the schema-changing program. This invention is not limited to this. A  
5 program storage medium storing the schema-changing program may be read by the client terminal CT and thus installed therein, to thereby execute the process of changing the schema.

[0153]

Furthermore, in the embodiment described above, there has been  
10 described the case where the control unit 100 of the update-information providing server 3 transmits the programs of the latest version and the schema-change instruction file SF as update information via the network 2 to the client terminal CT that has requested them, in accordance with the update-information providing program stored in the ROM 13 in advance. This invention is not limited to this.  
15 A program storage medium storing a communications program may be read by the update-information providing server 3 and thus installed therein, to thereby transmit the programs of the latest version and the schema-change instruction file SF to the client terminal CT.

[0154]

20 Further, in the above-described embodiment, there has been described the case where the client terminal CT as a communication terminal is constituted of the CPU 11 and the communication process unit 22 that are used as a requesting means and a receiving means, and the CPU 11 that is used as a program-updating means, a comparing means and a database-schema updating  
25 means. The invention is not limited to this. The communication terminal may be constituted with various circuit configurations of other types.

[0155]

Furthermore, in the embodiment described above, there has been described the case where the update-information providing server 3 as an  
30 updated-information providing apparatus is constituted of the hard disk drive 103 used as a storage means, and the control unit 100 and the data-communication

process unit 104 used as a transmitting means for transmitting to the client terminal CT the program and the schema-change instruction file SF, which are used as program update information and update information about the database schema, respectively. The present invention is not limited to this. Other various circuit configurations may constitute a program-providing apparatus.

[INDUSTRIAL APPLICABILITY]

[0156]

The present invention can be applied to a communication terminal that has a database.

[BRIEF DESCRIPTION OF THE DRAWINGS]

[0157]

[FIG. 1] A schematic diagram showing the overall configuration of an information providing system according to the present invention.

[FIG. 2] A schematic block diagram showing a configuration of a radio station.

[FIG. 3] A schematic block diagram illustrating a configuration of a related-information providing server.

[FIG. 4] A schematic block diagram illustrating a configuration of an update-information providing server.

[FIG. 5] A schematic block diagram showing a circuit configuration of a client terminal.

[FIG. 6] A schematic diagram explaining how a directory of contents is managed.

[FIG. 7] A schematic diagram showing a program module of the client terminal.

[FIG. 8] A schematic diagram illustrating a sequence of acquiring music information related to musical pieces contained in radio programs.

[FIG. 9] A schematic diagram representing a program configuration of the client terminal.

[FIG. 10] A schematic diagram outlining a schema-changing process.

[FIG. 11] A flowchart showing a schema-change procedure.

[FIG. 12] A schematic diagram representing a configuration of a schema-change instruction file.

[FIG. 13] A schematic diagram representing a configuration of a schema-change log file.

5 [FIG. 14] A schematic diagram illustrating an example of changing the schema on the basis of the schema-change instruction file.

[FIG. 15] A schematic diagram showing an outline of an exclusive control process in the schema-changing process.

10 [FIG. 16] An arrow chart representing an exclusive control procedure in the schema-changing process.

[FIG. 17] An arrow chart representing the exclusive control procedure performed when an application program is executed.

[DESCRIPTION OF REFERENCE NUMERALS]

[0158]

15 1...information providing system, 2...network, 3...update-information providing server, CT...client terminal, KS...related-information providing server, RS...radio station, SF...schema-change instruction file, SL...schema-change log file

【書類名】 図面  
【図 1】

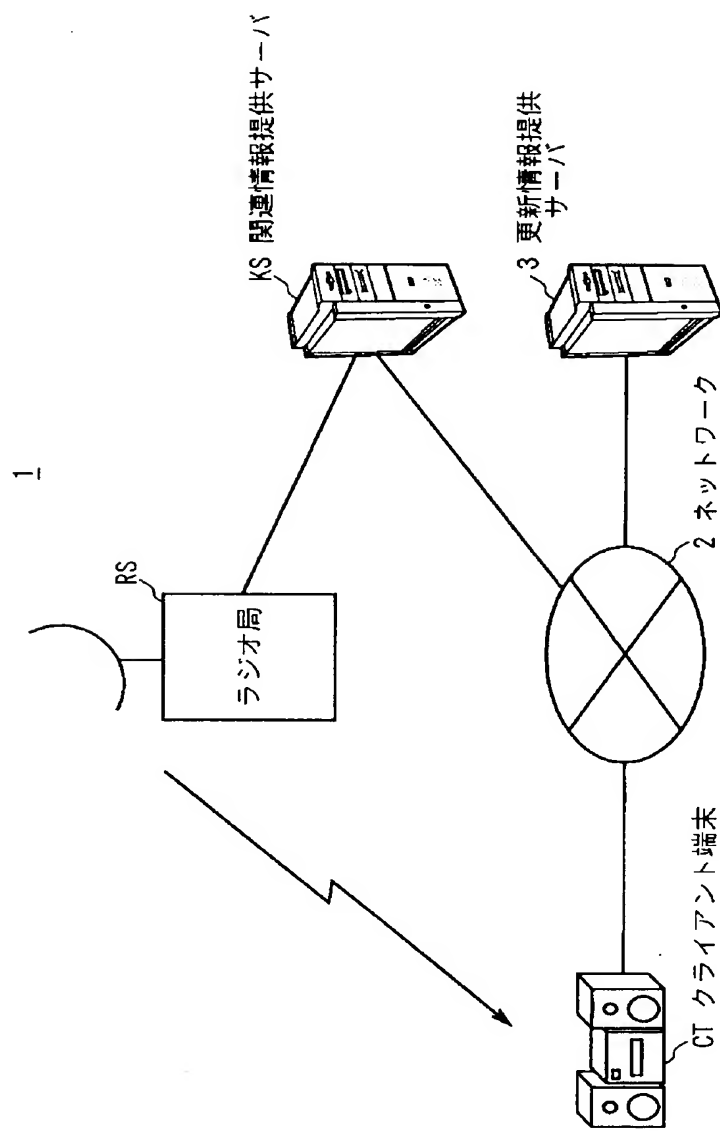


図 1 情報提供システムの全体構成

【図 2】

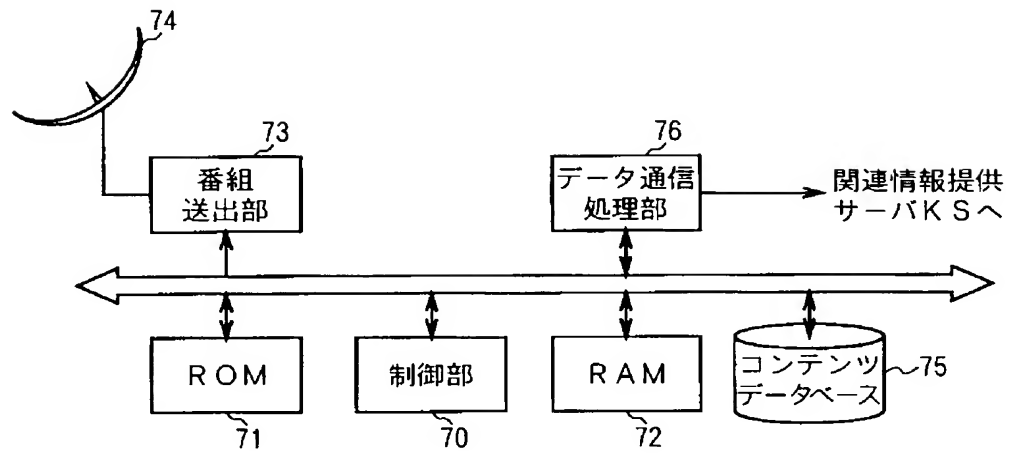


図 2 ラジオ局 R S の構成

【図 3】

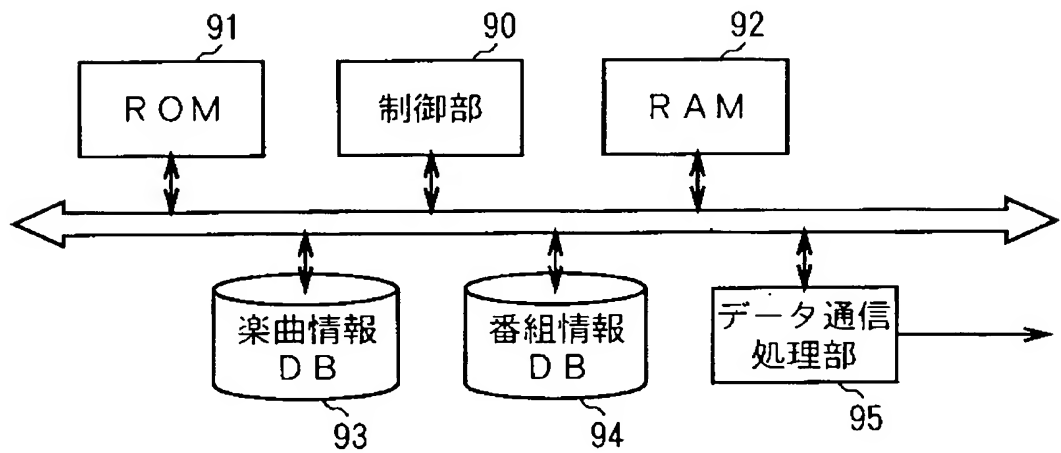


図 3 関連情報提供サーバの構成



【図 4】

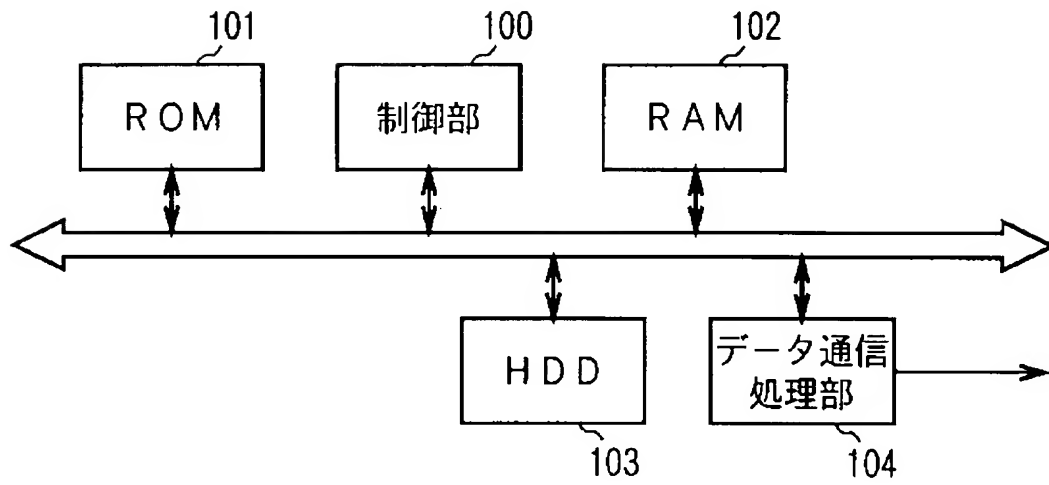


図 4 更新情報提供サーバの構成

【図 5】

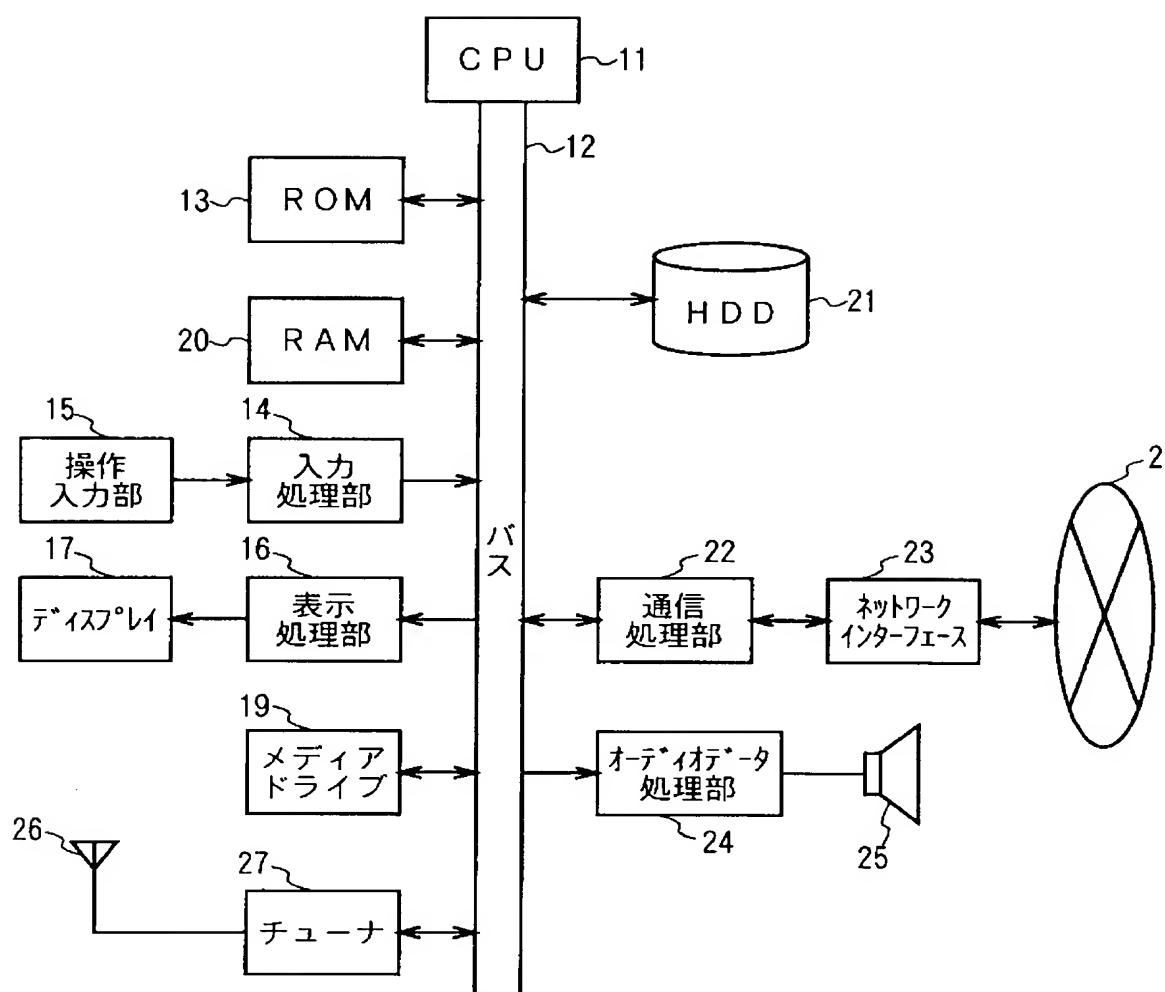


図 5 クライアント端末の回路構成

【図 6】

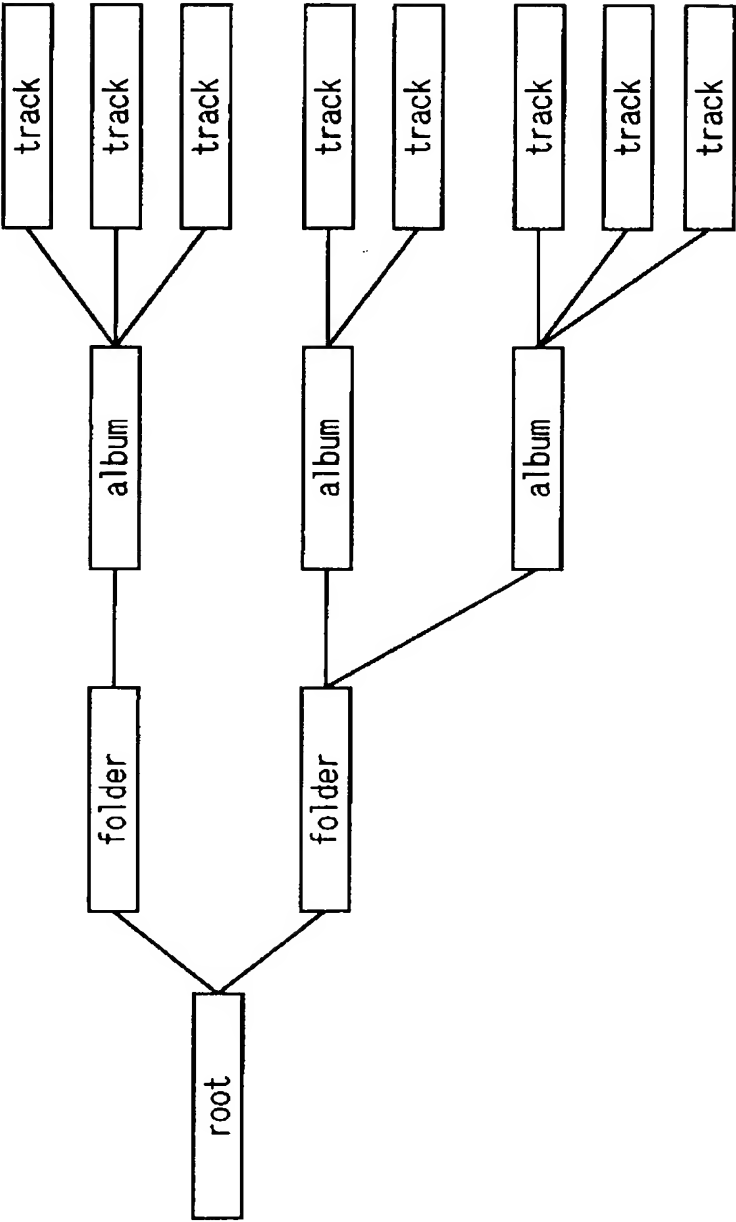


図 6 ディレクトリ構成

【図 7】

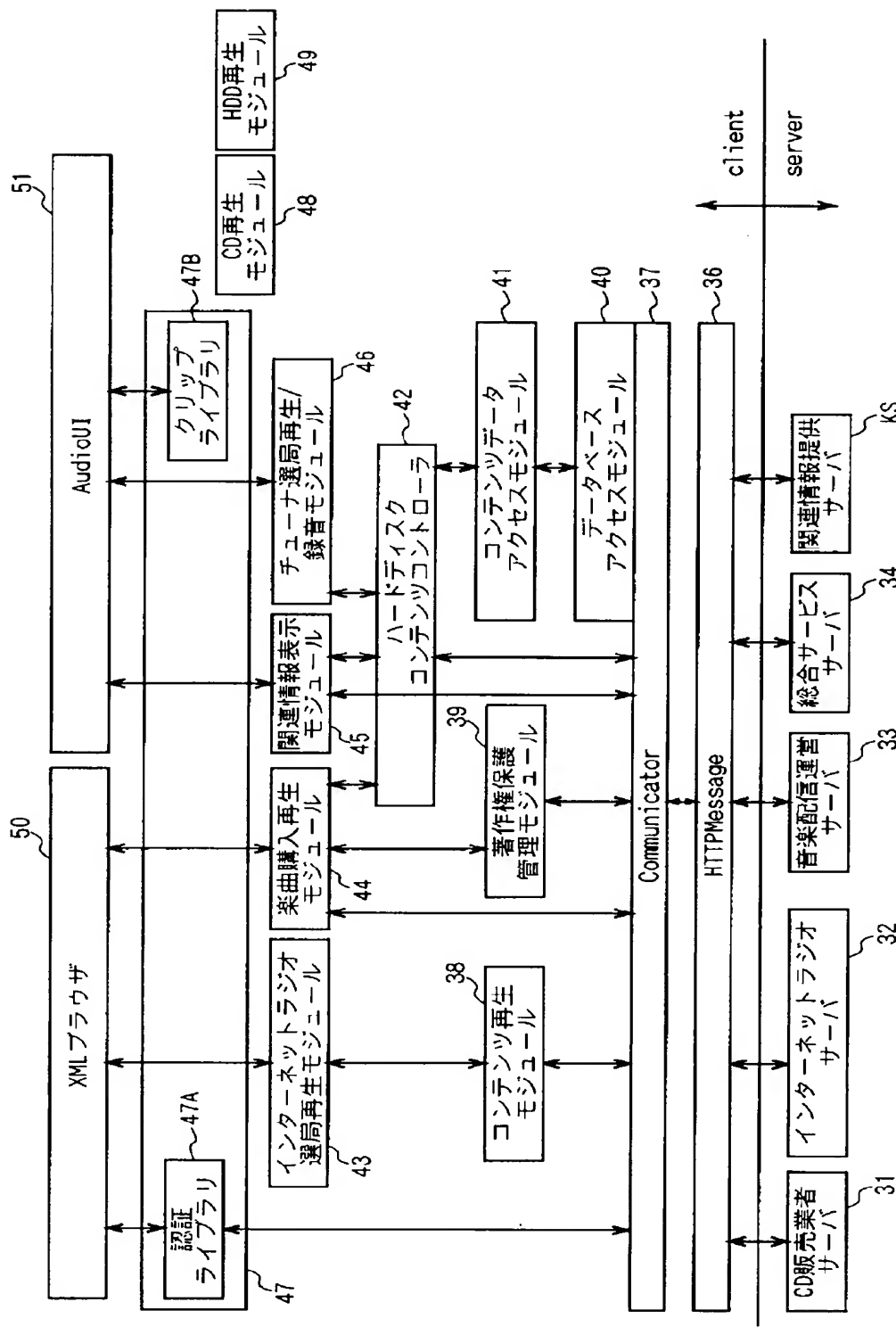


図 7 クライアント端末のプログラムモジュール

【図 8】

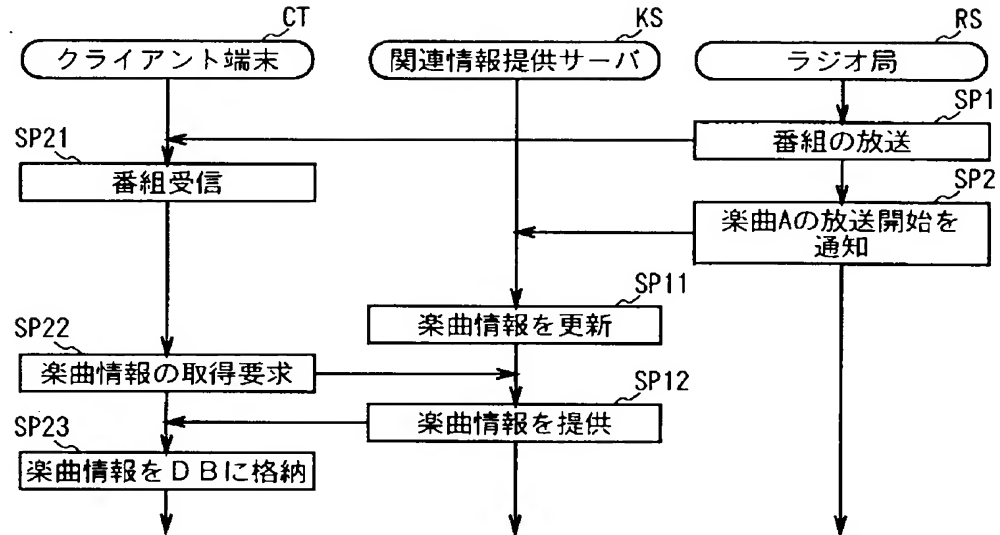


図 8 ラジオ番組の楽曲に関連した楽曲情報の取得処理シーケンス

【図 9】

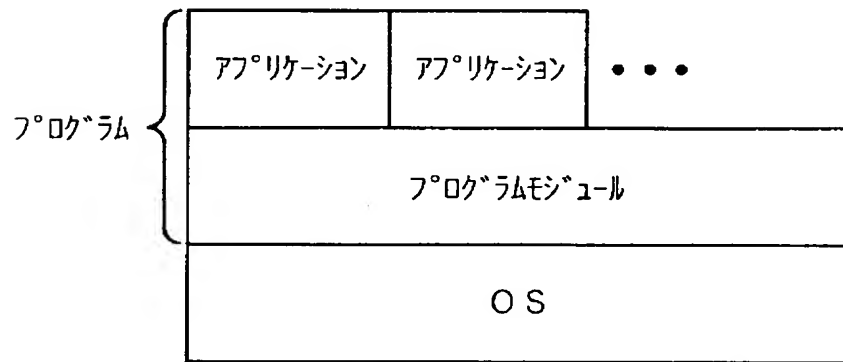


図 9 クライアント端末のプログラム構成

【図 10】

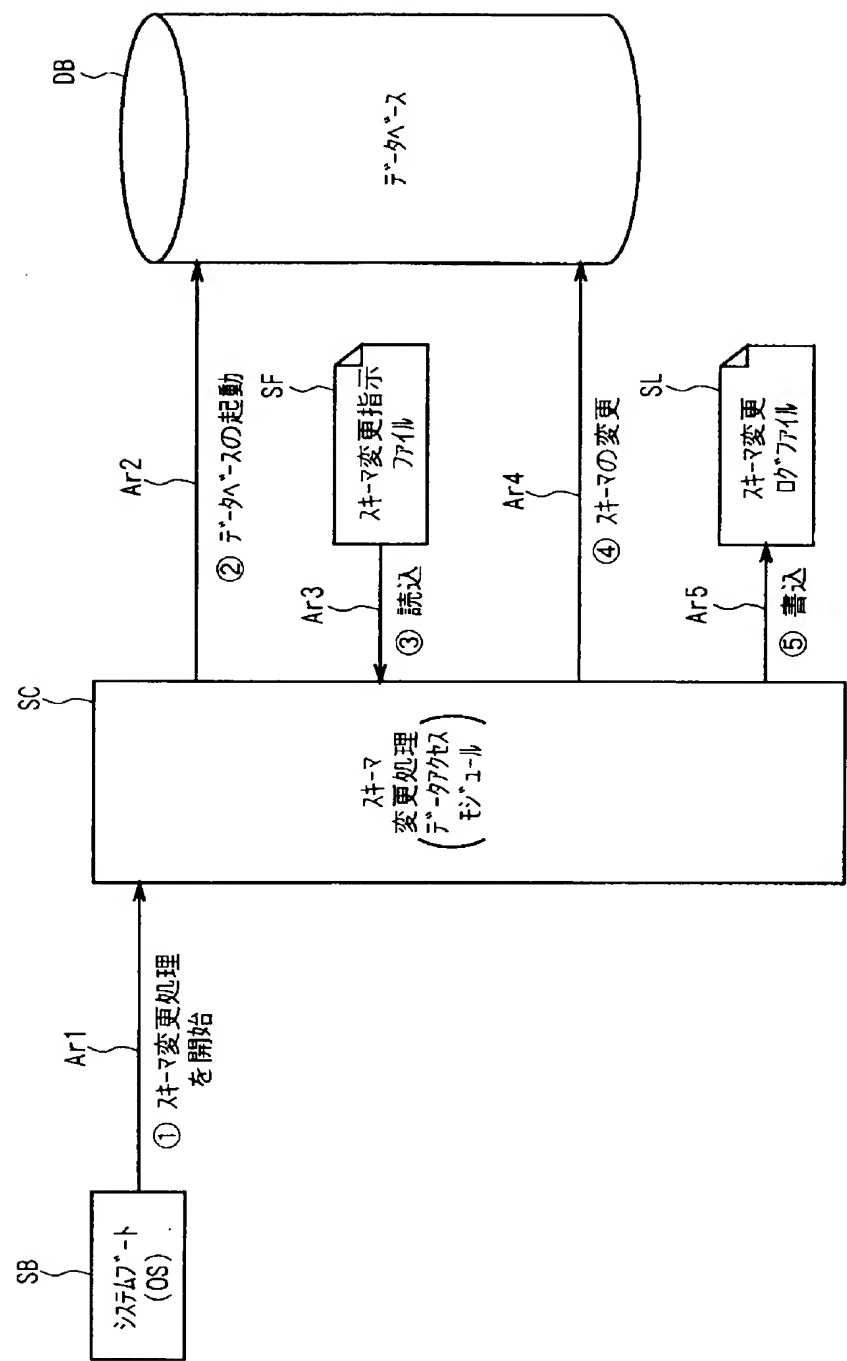


図 10 スキーマ変更処理の概要

【図 1 1】

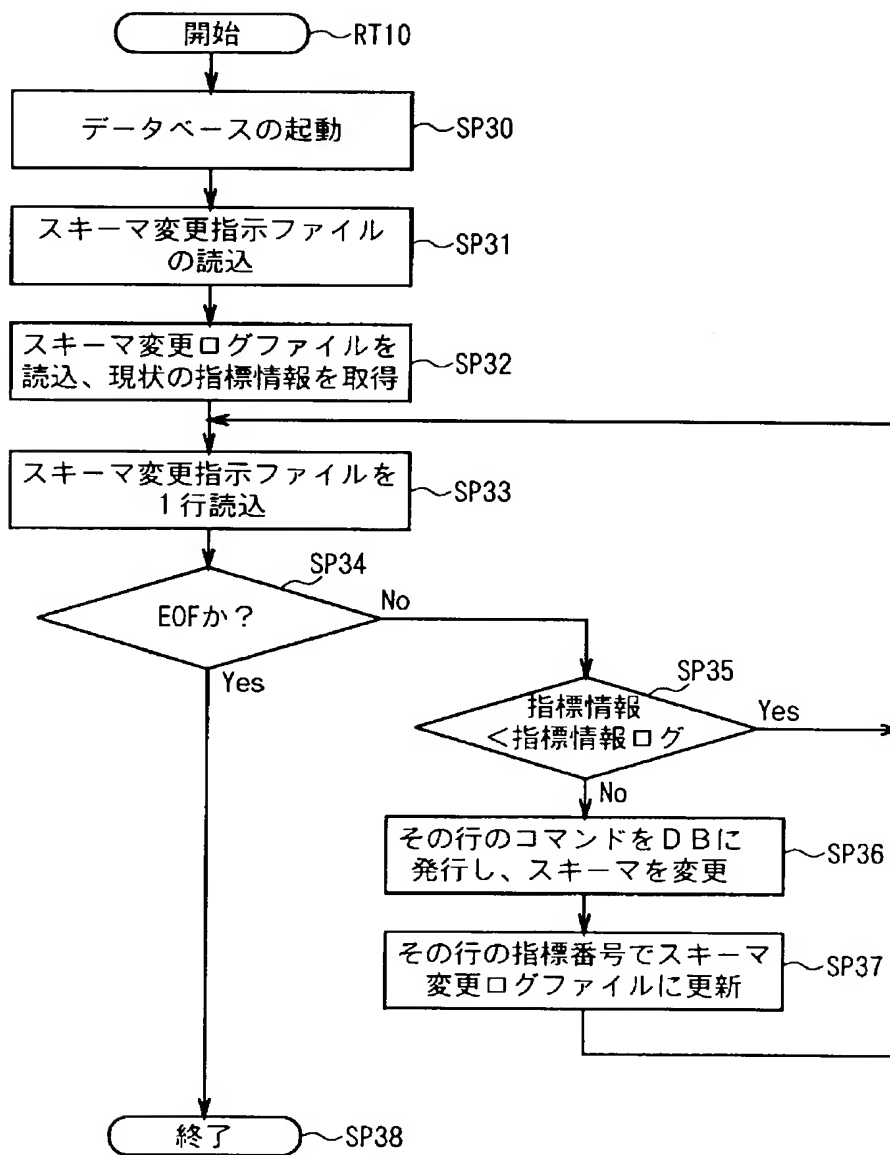


図 1 1 スキーマ変更処理手順

【図 1 2】

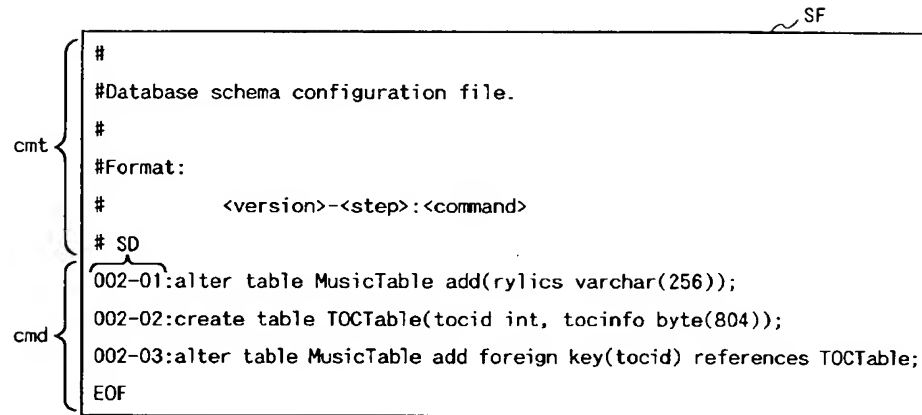


図 1 2 スキーマ変更指示ファイルの構成

【図 1 3】



図 1 3 スキーマ変更ログファイルの構成

【図 1 4】

5

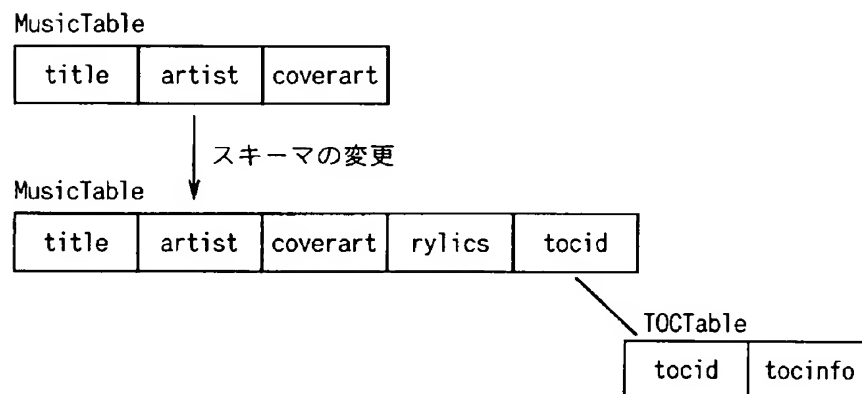


図 1 4 スキーマ変更指示ファイルに基づくスキーマの変更例



【図 15】

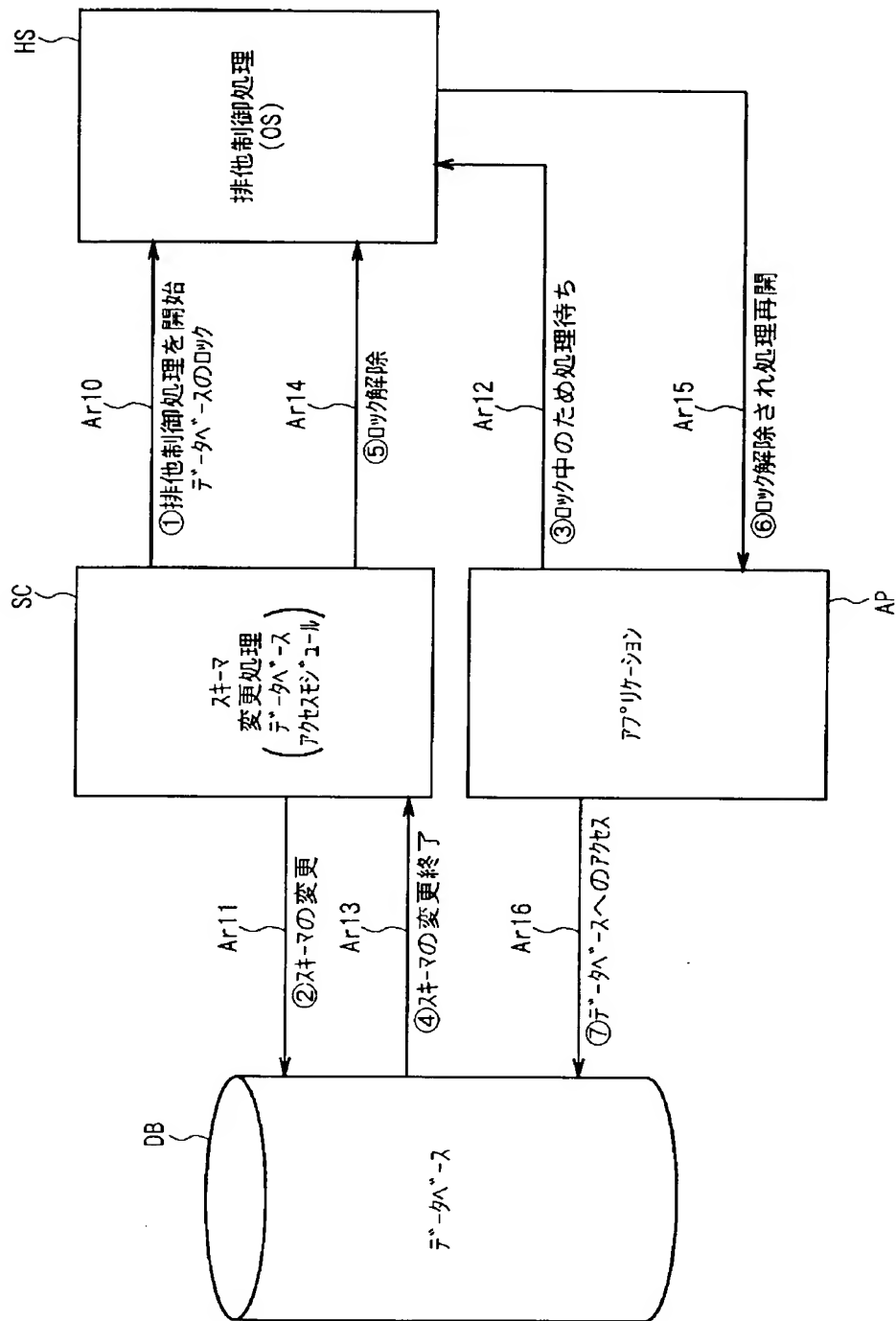


図 15 スキーマ変更処理時における排他制御処理の概要

【図 1 6】

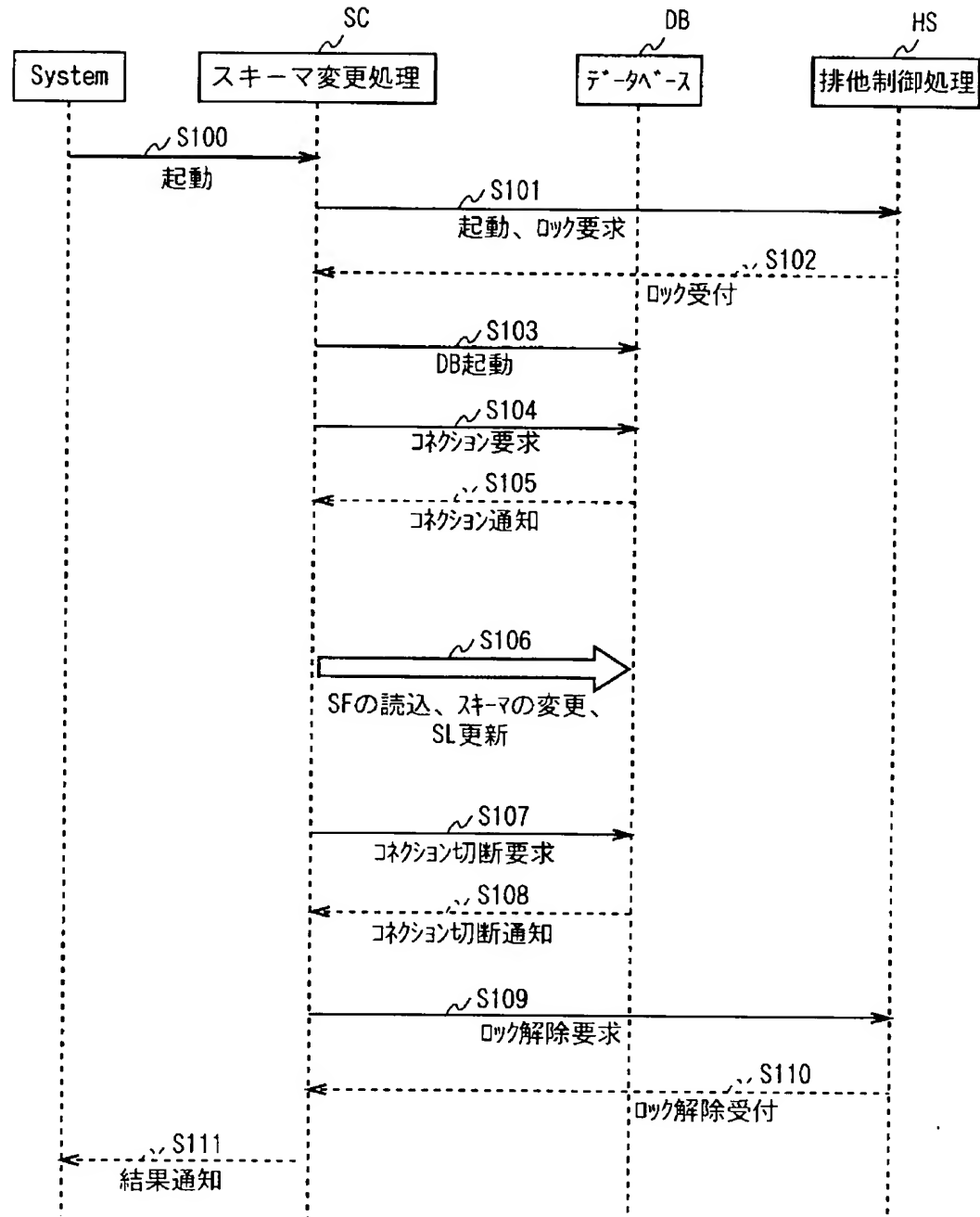


図 1 6 スキーマ変更処理時の排他制御処理手順

【図 1 7】

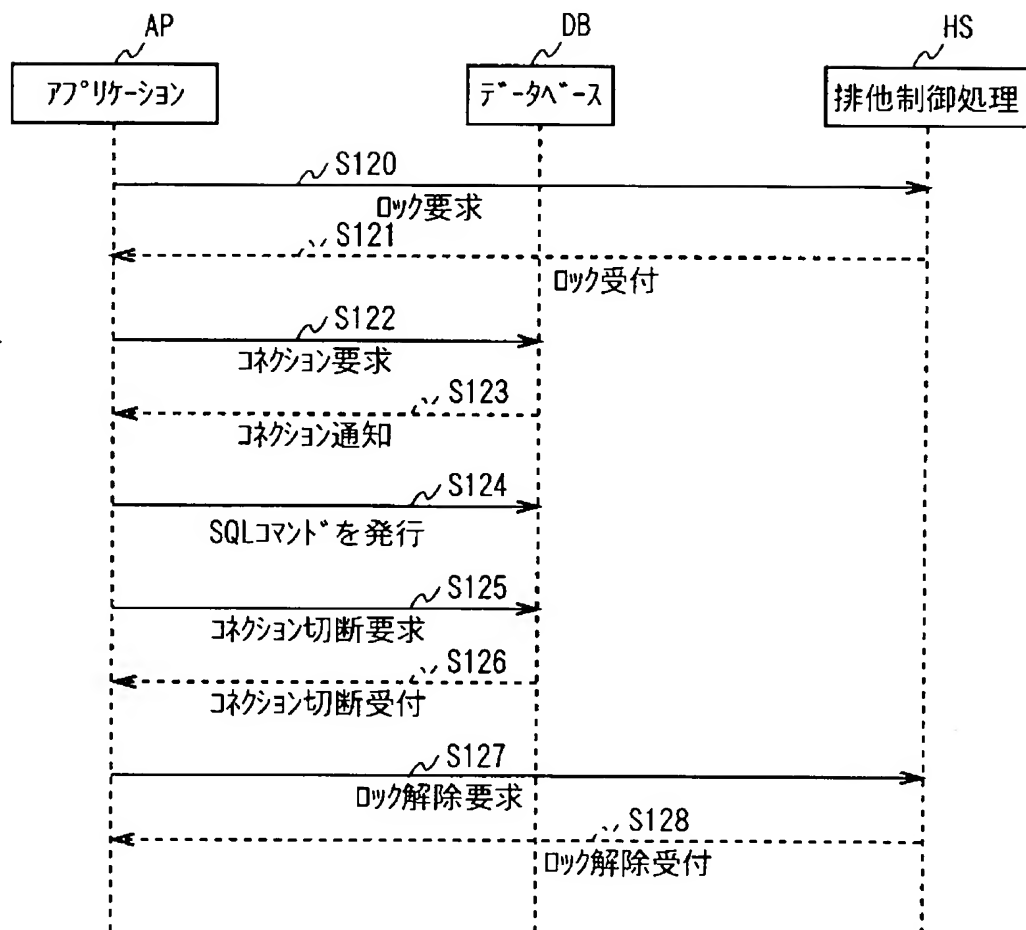


図 1 7 アプリケーション実行時の排他制御処理

【書類名】 図面 [NAME OF DOCUMENT] DRAWINGS

【図 1】 FIG. 1

2 Network  
5 3 Update-information providing server  
RS Radio station  
KS Related-information providing server  
CT Client terminal  
情報提供システムの全体構成 Overall configuration of information providing  
10 system

【図 2】 FIG. 2

70 Control unit  
73 Program transmitting unit  
15 75 Content database  
76 Data-communication process unit  
関連情報提供サーバ K S へ To related-information providing server KS  
ラジオ局 R S の構成 Configuration of radio station

20 【図 3】 FIG. 3

90 Control unit  
93 Music information database  
94 Program information database  
95 Data-communication process unit  
25 関連情報提供サーバの構成 Configuration of related-information providing  
server

【図 4】 FIG. 4

30 100 Control unit  
104 Data-communication process unit

更新情報提供サーバの構成      Configuration of update-information providing server

【図 5】      FIG. 5

- 5      12      Bus  
14      Input process unit  
15      Operation input unit  
16      Display process unit  
17      Display  
10      19      Media drive  
22      Communication process unit  
23      Network interface  
24      Audio data process unit  
27      Tuner  
15      クライアント端末の回路構成      Circuit configuration of client terminal

【図 6】      FIG. 6

ディレクトリ構成      Directory configuration

20      【図 7】      FIG. 7

- 31      CD seller server  
32      Internet radio server  
33      Music provider server  
34      Integrated service server  
25      38      Content playback module  
39      Copyright protection management module  
40      Database access module  
41      Content-data access module  
42      Hard-disk content controller  
30      43      Internet radio-station selection/playback module  
44      Musical-piece purchase/playback module

45      Related-information display module  
 46      Tuner-selection playback/recording module  
 47A     Authentication library  
 47B     Clip library  
 5      48      CD playback module  
 49      HDD playback module  
 50      XML browser  
 KS      Related-information providing server  
 クライアント端末のプログラムモジュール      Program module of client  
 10      terminal

【図 8】      FIG. 8

CT      Client terminal  
 KS      Related-information providing server  
 15      RS      Radio station  
 SP1     Broadcast program  
 SP2     Inform start of broadcasting musical piece A  
 SP11    Update music information  
 SP12    Provide music information  
 20      SP21    Receive program  
 SP22    Request acquisition of music information  
 SP23    Store music information in DB  
 ラジオ番組の楽曲に関連した楽曲情報の取得処理シーケンス      Sequence of  
 acquiring music information related to musical pieces in radio programs

25

【図 9】      FIG. 9

プログラム      Program  
 アプリケーション      Application  
 プログラムモジュール      Program module  
 30      クライアント端末のプログラム構成      Program configuration of client  
 terminal

【図 10】 FIG. 10

SB      System boot  
 SC      Schema-changing process (data access module)  
 5 DB      Database  
 SF      Schema-change instruction file  
 SL      Schema-change log file  
 Ar1     Start schema-changing process  
 Ar2     Activate database  
 10 Ar3     Read  
 Ar4     Change schema  
 Ar5     Write  
 スキーマ変更処理の概要      Outline of schema-changing process

15 【図 11】 FIG. 11

RT10    Start  
 SP30    Activate database  
 SP31    Read schema-change instruction file  
 SP32    Read schema-change log file and acquire current indicator information  
 20 SP33    Read one line of schema-change instruction file  
 SP34    EOF?  
 SP35    Indicator information < Indicator information log  
 SP36    Issue command of the line to DB and change schema  
 SP37    Update schema-change log file with indicator number of the line  
 25 SP38    End  
 スキーマ変更処理手順    Schema-change procedure

【図 12】 FIG. 12

スキーマ変更指示ファイルの構成      Configuration of schema-change  
 30 instruction file

【図 1 3】 FIG. 13

スキーマ変更ログファイルの構成 Configuration of schema-change log file

5 【図 1 4】 FIG. 14

スキーマの変更 Change of schema

スキーマ変更指示ファイルに基づくスキーマの変更例 Example of changing schema on the basis of schema-change instruction file

10 【図 1 5】 FIG. 15

DB Database

SC Schema-changing process (data access module)

AP Application

HS Exclusive control process

15 Ar10 Start exclusive control process

Lock database

Ar11 Change schema

Ar12 Wait because database remains locked

Ar13 End change of schema

20 Ar14 Unlock database

Ar15 Unlock database and restart process

Ar16 Access database

スキーマ変更処理時における排他制御処理の概要 Outline of exclusive control process in schema-changing process

25

【図 1 6】 FIG. 16

SC Schema-changing process

DB Database

HS Exclusive control process

30 S100 Activate

S101 Activate, and Request lock



S102 Accept lock  
S103 Activate DB  
S104 Request connection  
S105 Inform connection  
5 S106 Read SF, Change schema, and Update SL  
S107 Request disconnection  
S108 Inform disconnection  
S109 Request unlock  
S110 Accept unlock  
10 S111 Inform result  
スキーマ変更処理時の排他制御処理手順 Exclusive control procedure  
in schema-changing process

【図 17】 FIG. 17

15 AP Application  
DB Database  
HS Exclusive control process  
S120 Request lock  
S121 Accept lock  
20 S122 Request connection  
S123 Inform connection  
S124 Issue SQL command  
S125 Request disconnection  
S126 Accept disconnection  
25 S127 Request unlock  
S128 Accept unlock  
アプリケーション実行時の排他制御処理 Exclusive control procedure  
in executing application

[NAME OF DOCUMENT] ABSTRACT

[SUMMARY]

[OBJECT]

5 To realize a method of updating a database schema, which is capable of automatically update a database schema so that the database schema is of the latest version all times.

[MEANS FOR SOLVING]

10 In the present invention, a client terminal CT receives programs of the latest version and a schema-change instruction file SF from an update-information providing server 3 to upgrade the programs, and then compares an indicator information SD of the schema-change instruction file SF with an indicator information SDL of a schema-change log file SL to recognize the version of the present schema. If it is judged that the present schema needs to be upgraded, the schema is automatically changed based on the  
15 schema-change instruction file SF. Thus, the database schema can be automatically updated to be of the latest version all times.

[SELECTED DRAWING] Figure 11